400 comandos de sistemas GNU/Linux



Sumario

Introducción

Esta es una selección de 400 comandos de sistemas GNU/Linux para diversas gestiones y acciones a la hora de recabar información o realizar ciertas tareas.

La lista está confeccionada por el Grupo de Usuarios de Tecnologías Libres de Cuba (GUTL) y que está accesible en su wiki:

- https://gutl.jovenclub.cu/wiki/doku.php?id=/comandos/inicio

Esta recopilación en PDF está realizada como método de consulta cuando no tengamos acceso a internet o para guardarla en nuestro equipo y tenerla a mano para una consulta rápida.

Espero que la disfrutes y la compartas.

Victorhck

Información del sistema

1. `arch` → mostrar la arquitectura de la máquina (1).
2. `uname -m` → mostrar la arquitectura de la máquina (2).
3. `uname -r` → mostrar la versión del kernel usado.
4. `uname -a` → mostrar la información completa.
5. `lsb_release -a` → mostrar la información completa de la distribución.
6. `cat /etc/issue` → mostrar el nombre de la distribución
7. `dmidecode -q` → mostrar los componentes (hardware) del sistema.
8. `hdparm -i /dev/hda` → mostrar las características de un disco duro.
9. `hdparm -tT /dev/sda` → realizar prueba de lectura en un disco duro.
10. `cat /proc/cpuinfo` → mostrar información de la CPU.
11. `grep -c ^processor /proc/cpuinfo` → mostrar número de procesadores.
12. `cat /proc/interrupts` → mostrar las interrupciones.
13. `cat /proc/meminfo` → verificar el uso de memoria.
14. `cat /proc/swaps` → mostrar ficheros swap.
15. `cat /proc/version` → mostrar la versión del kernel.
16. `cat /proc/net/dev` → mostrar adaptadores de red y estadísticas.
17. `cat /proc/mounts` → mostrar el sistema de ficheros montado.
18. `lscpu` → mostrar información sobre el microprocesador.
19. `lspci -tv` → mostrar los dispositivos PCI.
20. `lsusb -tv` → mostrar los dispositivos USB.
21. `lshw` → listar el hardware.
22. `discover` → listar el hardware.
23. `date` → mostrar la fecha del sistema.
24. `cal 2011` → mostrar el almanaque de 2011.
25. `cal 07 2011` → mostrar el almanaque para el mes julio de 2011.
26. `date 041217002011.00` → colocar (declarar, ajustar) fecha y hora.
27. `clock -w` → guardar los cambios de fecha en la BIOS.
28. `blkid` → mostrar información (nombre, etiqueta, UUID, tipo de partición) sobre los dispositivos de bloque (discos rígidos, etc.)

Apagar, reiniciar o cerrar sesión

1. `shutdown -h now` → apagar el sistema (1).
2. `init 0` → apagar el sistema (2).
3. `telinit 0` → apagar el sistema (3).
4. `halt` → apagar el sistema (4).
5. `poweroff` → apagar sistema (5).
6. `shutdown -h hours:minutes &` → apagado planificado del sistema.
7. `shutdown -c` → cancelar un apagado planificado del sistema.
8. `shutdown -r now` → reiniciar (1).
9. `reboot` → reiniciar (2).

10. `logout` → cerrar sesión.
11. `skill nombre_de_usuario` → cerrar sesión (2)[1)]
12. `exit` → salir del intérprete de comandos (si solo hay uno, equivale a cerrar sesión).

Gestionar archivos y directorios

1. `cd` → ir al directorio personal.
2. `cd /home` → cambiar al directorio "/home".
3. `cd ..` → retroceder un nivel.
4. `cd ../..` → retroceder 2 niveles.
5. `cd ~user1` → ir al directorio user1.
6. `cd -` → ir (regresar) al directorio anterior.
7. `pwd` → mostrar el camino del directorio actual.
8. `ls` → listar el contenido de un directorio.
9. `ls -F` → listar el contenido de un directorio (distinguiendo los directorios con una barra)
10. `ls -l` → listar el contenido de un directorio, mostrando los detalles.
11. `ls -lh` → listar el contenido de un directorio, mostrando los detalles (y el tamaño en un formato "humanizado").
12. `ls -a` → listar el contenido de un directorio, incluendo los ficheros ocultos.
13. `ls *[0-9]` → listar los ficheros y carpetas que contienen números.
14. `ls -laR | less` → listar recursivamente el contenido del directorio actual y todos los subdirectorios y archivos, incluyendo los ocultos, separados por página.
15. `tree` → mostrar los ficheros y carpetas en forma de árbol comenzando por la raíz.(1)
16. `lstree` → mostrar los ficheros y carpetas en forma de árbol comenzando por la raíz.(2)
17. `mkdir dir1` → crear un directorio de nombre 'dir1'.
18. `mkdir dir1 dir2` → crear dos directorios a la vez (en la ubicación actual).
19. `mkdir -p /tmp/dir1/dir2` → crear una estructura de directorios, si no existe.
20. `rm file1` → eliminar el archivo 'file1'.
21. `rm -f file1` → eliminar el archivo 'file1' en modo forzado.
22. `rmdir dir1` → borrar el directorio 'dir1'.
23. `rm -rf dir1` → eliminar recursivamente y en modo forzado el directorio 'dir1' con todo lo que contenga.
24. `rm -rf dir1 dir2` → borrar dos directorios con su contenido de forma recursiva.
25. `mv dir1 new_dir` → renombrar o mover un fichero o carpeta (directorio).
26. `cp file1 destino/` → copiar un fichero al destino elegido.
27. `cp file1 file2 destino/` → copiar a la vez dos ficheros a un mismo directorio.
28. `cp file1 file2` → copiar file1 en file2.
29. `cp dir /* .` → copiar todos los ficheros de un directorio dentro del directorio de trabajo actual.
30. `cp -a /tmp/dir1 .` → copiar un directorio dentro del directorio actual de trabajo.
31. `cp -a dir1` → copiar un directorio.
32. `cp -a dir1 dir2` → copiar dos directorio al unísono.
33. `ln -s file1 lnk1` → crear un enlace simbólico al fichero o directorio.

34. `ln file1 lnk1` → crear un enlace físico al fichero o directorio.
35. `touch file1` → actualizar la fecha de modificación de file1, o crearlo si no existe.
36. `touch -t 0712250000 file1` → modificar el tiempo real (tiempo de creación) de un fichero o directorio.
37. `file file1` → salida (volcado en pantalla) del tipo mime de un fichero texto.
38. `iconv -l` → listas de cifrados conocidos.
39. `iconv -f fromEncoding -t toEncoding inputFile > outputFile` → crea una nueva forma del fichero de entrada asumiendo que está codificado en fromEncoding y convirtiéndolo a ToEncoding.

Encontrar archivos

1. `find / -name file1` → buscar fichero y directorio a partir de la raíz del sistema.
2. `find / -user user1` → buscar ficheros y directorios pertenecientes al usuario “user1”.
3. `find /home/user1 -name \*.bin` → buscar ficheros con extensión ”. bin” dentro del directorio `/ home/user1`.
4. `find /usr/bin -type f -atime +100` → buscar ficheros binarios no usados en los últimos 100 días.
5. `find /usr/bin -type f -mtime -10` → buscar ficheros creados o cambiados dentro de los últimos 10 días.
6. `find / -name \*.rpm -exec chmod 755 '{}' \;` → buscar ficheros con extensión ”.rpm” y modificar permisos.
7. `find . -type f -print0 | xargs -0 chmod 644` → modificar recursivamente los permisos a todos los ficheros bajo el directorio actual.[2]
8. `find / -xdev -name \*.rpm` → Buscar ficheros con extensión '.rpm' ignorando los dispositivos removibles como cdrom, pen-drive, etc.…
9. `find . -maxdepth 1 -name *.jpg -print -exec convert "{}" -resize 80×60 "thumbs/{}" \;` → agrupar ficheros redimensionados en el directorio actual y enviarlos a directorios en vistas de miniaturas (requiere convertir desde Imagemagick).
10. `find /tmp/dir1 -depth -regextype posix-extended -regex '.*(\s+|:+|\\+|>+|<+|"+|\*+|\?+|\|+).*' -execdir rename 's/(\s+|:+|\\+|>+|<+|”+|\*+|\?+|\|+)/_/g' "{}" \;` → renombrar recursivamente todos los directorios y ficheros bajo `/tmp/dir1`, cambiando los espacios y otros caracteres extraños por guiones bajos.
11. `locate \*.ps` → encuentra ficheros con extensión ”.ps” ejecutados primeramente con el command `updatedb`.
12. `whereis halt` → mostrar la ubicación de un fichero binario, de ayuda o fuente. En este caso pregunta dónde está el comando `halt`.
13. `which comando` → mostrar la ruta completa a un comando.

Montando un sistema de ficheros

1. `mount /dev/hda2 /mnt/hda2` → montar un disco llamado hda2. Verifique primero la existencia del directorio '/ mnt/hda2'; si no está, debe crearlo.
2. `umount /dev/hda2` → desmontar un disco llamado hda2. (Antes es necesario salir del punto '/mnt/hda2'.
3. `fuser -km /mnt/hda2` → forzar el desmontaje cuando el dispositivo está ocupado.
4. `umount -n /mnt/hda2` → correr el desmontaje sin leer el fichero /etc/mtab. Útil cuando el fichero es de solo lectura o el disco duro está lleno.
5. `mount /dev/fd0 /mnt/floppy` → montar un disco flexible (floppy).
6. `mount /dev/cdrom /mnt/cdrom` → montar un cdrom / dvdrom.
7. `mount /dev/hdc /mnt/cdrecorder` → montar un cd regrabable o un dvdrom.
8. `mount /dev/hdb /mnt/cdrecorder` → montar un cd regrabable / dvdrom (un dvd).
9. `mount -t udf,iso9660 -o loop file.iso /mnt/cdrom` → montar un fichero de imagen de un medio óptico (como un CD o DVD en formato ISO).
10. `mount -t vfat /dev/hda5 /mnt/hda5` → montar un sistema de ficheros FAT32.
11. `mount -t ntfs-3g /dev/hda5 /mnt/hda5` → montar un sistema de ficheros NTFS.
12. `mount /dev/sda1 /mnt/usbdisk` → montar un usb pen-drive o una memoria (sin especificar el tipo de sistema de ficheros).

Espacio en disco

1. `df -h` → mostrar una lista de las particiones montadas.
2. `ls -lSr | more` → mostrar el tamaño de los ficheros y directorios ordenados por tamaño.
3. `du -sh dir1` → Estimar el espacio usado por el directorio 'dir1'.
4. `du -sk * | sort -nr` → mostrar en orden descendente el tamaño de los ficheros y subdirectorios en la ubicación actual, en KiB.
5. `du -h --max-depth=1 | sort -nr` → mostrar en orden descendente el tamaño de todos los subdirectorios en la ubicación actual, usando unidades de medida adaptables.
6. `du -k --max-depth=1 | sort -k1 -nr | awk '{printf "%.3f GiB\t %s\n", $1/(1048576), $2}' | less` → mostrar en orden descendente el tamaño de los directorios y archivos, en GiB.
7. `rpm -q -a --qf '%10{SIZE}t%{NAME}n' | sort -k1,1n` → mostrar el espacio usado por los paquetes rpm instalados organizados por tamaño (Fedora, Redhat y otros).
8. `dpkg-query -W -f='${Package}\t${Installed-Size}\n' | sort -k 2 -nr | grep -v deinstall | head -n 25 | awk '{printf "%.3f \t %s\n", $2/(1024), $1}'` → mostrar (en Debian o derivadas) un listado con los 25 paquetes instalados que más espacio consumen (en orden descendente).

Usuarios y grupos

1. `groupadd nombre_del_grupo` → crear un nuevo grupo.

2. `groupdel nombre_del_grupo` → borrar un grupo.
3. `groupmod -n nuevo_nombre_del_grupo viejo_nombre_del_grupo` → renombrar un grupo.
4. `adduser usuario1` → Crear un nuevo usuario, mediante un asistente.
5. `useradd -c "Nombre Apellido" -g admin -d /home/usuario1 -s /bin/bash usuario1` → Crear un nuevo usuario perteneciente al grupo "admin".
6. `useradd usuario1` → crear un nuevo usuario.
7. `userdel -r usuario1` → borrar un usuario, eliminando su directorio Home.
8. `usermod -c "Usuario de " -g system -d /ftp/usuario1 -s /sbin/nologin usuario1` → cambiar los atributos de un usuario.
9. `usermod -aG plugdev,dialout,pip user1` → agregar el usuario user1 a otros grupos existentes, para incrementar sus permisos (en este caso, agregar la posibilidad de conectar dispositivos, configurar y utilizar el modem)
10. `passwd` → cambiar contraseña.
11. `passwd usuario1` → cambiar la contraseña de 'usuario1' (solamente ejecutable como superusuario).
12. `chage -E 2011-12-31 user1` → colocar un plazo para la contraseña del usuario. En este caso dice que la clave expira el 31 de diciembre de 2011.
13. `pwck` → chequear la sintaxis correcta el formato de fichero de '/etc/passwd' y la existencia de usuarios.
14. `grpck` → chequear la sintaxis correcta y el formato del fichero '/etc/group' y la existencia de grupos.
15. `newgrp grupo1` → registra a un nuevo grupo para cambiar el grupo predeterminado de los ficheros creados recientemente.

Permisos en ficheros (usar "+" para colocar permisos y "-" para eliminar)

1. `ls -lh` → Mostrar permisos.
2. `ls /tmp | pr -T5 -W$COLUMNS` → dividir la terminal en 5 columnas.
3. `chmod ugo+rwx directory1` → colocar permisos de lectura ®, escritura (w) y ejecución(x) al propietario (u), al grupo (g) y a otros (o) sobre el directorio 'directory1'.
4. `chmod go-rwx directory1` → quitar permiso de lectura ®, escritura (w) y (x) ejecución al grupo (g) y otros (o) sobre el directorio 'directory1'.
5. `chown user1 file1` → cambiar el dueño de un fichero.
6. `chown -R user1 directory1` → cambiar el propietario de un directorio y de todos los ficheros y directorios contenidos dentro.
7. `chgrp group1 file1` → cambiar grupo de ficheros.
8. `chown user1:group1 file1` → cambiar usuario y el grupo propietario de un fichero.
9. `find / -perm -u+s` → visualizar todos los ficheros del sistema con SUID configurado.
10. `chmod u+s /bin/file1` → colocar el bit SUID en un fichero binario. El usuario que corriendo ese fichero adquiere los mismos privilegios como dueño.
11. `chmod u-s /bin/file1` → deshabilitar el bit SUID en un fichero binario.

12. `chmod g+s /home/public` → colocar un bit SGID en un directorio –similar al SUID pero por directorio.
13. `chmod g-s /home/public` → desabilitar un bit SGID en un directorio.
14. `chmod o+t /home/public` → colocar un bit STIKY en un directorio. Permite el borrado de ficheros solamente a los dueños legítimos.
15. `chmod o-t /home/public` → desabilitar un bit STIKY en un directorio.

Atributos especiales en ficheros (usar "+" para colocar permisos y "-" para eliminar)

1. `chattr +a file1` → permite escribir abriendo un fichero solamente modo append.
2. `chattr +c file1` → permite que un fichero sea comprimido / descomprimido automaticamente.
3. `chattr +d file1` → asegura que el programa ignore borrar los ficheros durante la copia de seguridad.
4. `chattr +i file1` → convierte el fichero en inmutable o invariable, por lo que no puede ser eliminado, alterado, renombrado, ni enlazado.
5. `chattr +s file1` → permite que un fichero sea borrado de forma segura.
6. `chattr +S file1` → asegura que un fichero sea modificado, los cambios son escritos en modo synchronous como con sync.
7. `chattr +u file1` → te permite recuperar el contenido de un fichero aún si este está cancelado.
8. `lsattr` → mostrar atributos especiales.

Archivos y ficheros comprimidos

1. `7za a -mx=9 -ms=on -mhe=on -p archivocomprimido directorio1 archivo1 archivo2` → comprimir un directorio y dos archivos en formato 7zip, con compresión sólida máxima, y protección por contraseña (la extensión 7z se agrega automáticamente).
2. `7za x archivocomprimido.7z` → extraer un archivo comprimido en 7zip (7zip también permite descomprimir otros formatos, como por ejemplo, zip).
3. `bunzip2 file1.bz2` → descomprime in fichero llamado 'file1.bz2'.
4. `bzip2 file1` → comprime un fichero llamado 'file1'.
5. `gunzip file1.gz` → descomprime un fichero llamado 'file1.gz'.
6. `gzip file1` → comprime un fichero llamado 'file1'.
7. `gzip -9 file1` → comprime con compresión máxima.
8. `rar a file1.rar test_file` → crear un fichero rar llamado 'file1.rar'.
9. `rar a file1.rar file1 file2 dir1` → comprimir 'file1', 'file2' y 'dir1' simultáneamente.
10. `rar x file1.rar` → descomprimir archivo rar.
11. `unrar x file1.rar` → descomprimir archivo rar.
12. `tar -cvf archive.tar file1` → crear un tarball descomprimido.

13. `tar -cvf archive.tar file1 file2 dir1` → crear un archivo conteniendo 'file1', 'file2' y'dir1'.
14. `tar -tf archive.tar` → mostrar los contenidos de un archivo.
15. `tar -xvf archive.tar` → extraer un tarball (si el archivo además está comprimido con gzip, bzip2 o xz, descomprimirlo automáticamente).
16. `tar -xvf archive.tar -C /tmp` → extraer un tarball en /tmp.
17. `tar -cjvf archive.tar.bz2 dir1` → crear un tarball comprimido en bzip2.
18. `tar -xjvf archive.tar.bz2` → descomprimir un archivo tar comprimido en bzip2
19. `tar -cJvf archive.tar.xz dir1` → crear un tarball comprimido en xz.
20. `XZ_OPT=-9e tar -cJvf archive.tar.xz dir1` → crear un tarball comprimido en xz (con máxima compresión).
21. `tar -xJvf archive.tar.xz` → descomprimir un archivo tar comprimido en xz.
22. `tar -czvf archive.tar.gz dir1` → crear un tarball comprimido en gzip.
23. `tar -I pigz -cf archive.tar.gz dir1` → crear un tarball comprimido en gzip, pero utilizando `pigz`, que comprime en paralelo aprovechando todos los núcleos de los microprocesadores del equipo.
24. `GZIP=-9 tar -czvf archive.tar.gz dir1` → crear un tarball comprimido en gzip (con máxima compresión).
25. `tar -xzvf archive.tar.gz` → descomprimir un archive tar comprimido en gzip.
26. `zip file1.zip file1` → crear un archivo comprimido en zip.
27. `zip -r file1.zip file1 file2 dir1` → comprimir, en zip, varios archivos y directorios de forma simultánea.
28. `unzip file1.zip` → descomprimir un archivo zip.

Paquetes rpm (Red Hat, Fedora y similares)

1. `rpm -ivh package.rpm` → instalar un paquete rpm.
2. `rpm -ivh --nodeeps package.rpm` → instalar un paquete rpm ignorando las peticiones de dependencias.
3. `rpm -U package.rpm` → actualizar un paquete rpm sin cambiar la configuración de los ficheros.
4. `rpm -F package.rpm` → actualizar un paquete rpm solamente si este está instalado.
5. `rpm -e package_name.rpm` → eliminar un paquete rpm.
6. `rpm -qa` → mostrar todos los paquetes rpm instalados en el sistema.
7. `rpm -qa | grep httpd` → mostrar todos los paquetes rpm con el nombre “httpd”.
8. `rpm -qi package_name` → obtener información en un paquete específico instalado.
9. `rpm -qg "System Environment/Daemons"` → mostar los paquetes rpm de un grupo software.
10. `rpm -ql package_name` → mostrar lista de ficheros dados por un paquete rpm instalado.
11. `rpm -qc package_name` → mostrar lista de configuración de ficheros dados por un paquete rpm instalado.
12. `rpm -q package_name --whatrequires` → mostrar lista de dependencias solicitada para un paquete rpm.

13. `rpm -q package_name --whatprovides` → mostar la capacidad dada por un paquete rpm.
14. `rpm -q package_name --scripts` → mostrar los scripts comenzados durante la instalación /eliminación.
15. `rpm -q package_name --changelog` → mostar el historial de revisions de un paquete rpm.
16. `rpm -qf /etc/httpd/conf/httpd.conf` → verificar cuál paquete rpm pertenece a un fichero dado.
17. `rpm -qp package.rpm -l` → mostrar lista de ficheros dados por un paquete rpm que aún no ha sido instalado.
18. `rpm --import /media/cdrom/RPM-GPG-KEY` → importar la firma digital de la llave pública.
19. `rpm --checksig package.rpm` → verificar la integridad de un paquete rpm.
20. `rpm -qa gpg-pubkey` → verificar la integridad de todos los paquetes rpm instalados.
21. `rpm -V package_name` → chequear el tamaño del fichero, licencias, tipos, dueño, grupo, chequeo de resumen de MD5 y última modificación.
22. `rpm -Va` → chequear todos los paquetes rpm instalados en el sistema. Usar con cuidado.
23. `rpm -Vp package.rpm` → verificar un paquete rpm no instalado todavía.
24. `rpm2cpio package.rpm | cpio --extract --make-directories *bin` → extraer fichero ejecutable desde un paquete rpm.
25. ``rpm -ivh /usr/src/redhat/RPMS/`arch`/package.rpm`` → instalar un paquete construido desde una fuente rpm.
26. `rpmbuild --rebuild package_name.src.rpm` → construir un paquete rpm desde una fuente rpm.

Actualizador de paquetes yum (Fedora, Redhat y otros)

1. `yum install package_name` → descargar e instalar un paquete rpm.
2. `yum localinstall package_name.rpm` → este instalará un RPM y tratará de resolver todas las dependencies para ti, usando tus repositorios.
3. `yum update` → actualizar todos los paquetes rpm instalados en el sistema.
4. `yum update package_name` → modernizar / actualizar un paquete rpm.
5. `yum remove package_name` → eliminar un paquete rpm.
6. `yum list` → listar todos los paquetes instalados en el sistema.
7. `yum search package_name` → Encontrar un paquete en repositorio rpm.
8. `yum clean packages` → limpiar un caché rpm borrando los paquetes descargados.
9. `yum clean headers` → eliminar todos los ficheros de encabezamiento que el sistema usa para resolver la dependencia.
10. `yum clean all` → eliminar desde los paquetes caché y ficheros de encabezado.

Gestión de paquetes deb (Debian, Ubuntu y otros)

1. `dpkg -i elpaquete.deb` → instalar / actualizar un paquete.
2. `dpkg -r elpaquete` → eliminar un paquete deb del sistema.

3. `dpkg -l` → mostrar todos los paquetes deb instalados en el sistema.
4. `dpkg -l | grep httpd` → mostrar todos los paquetes deb con el nombre “httpd”
5. `dpkg -s elpaquete` → obtener información en un paquete específico instalado en el sistema.
6. `dpkg -L elpaquete` → mostar lista de ficheros utilizados por un paquete instalado en el sistema.
7. `dpkg -c elpaquete.deb` → mostrar contenido de un paquete (no necesariamente instalado).
8. `dpkg -S /bin/ping` → verificar a qué paquete pertenece un fichero dado.

Actualizador de paquetes apt y aptitude (Debian, Ubuntu y otros)

1. `apt-get install package_name` → instalar / actualizar un paquete deb.
2. `apt-cdrom install package_name` → instalar / actualizar un paquete deb desde un cdrom.
3. `apt-get update` → actualizar la lista de paquetes.
4. `apt-get upgrade` → actualizar todos los paquetes instalados.
5. `apt-get remove package_name` → eliminar un paquete deb del sistema.
6. `apt-get check` → verificar la correcta resolución de las dependencias.
7. `apt-get clean` → limpiar cache desde los paquetes descargados.
8. `apt-cache search searched-package` → retorna lista de paquetes que corresponde a la serie «paquetes buscados».
9. `aptitude search paquete` → busca un paquete por el nombre.
10. `aptitude search ~dpaquete` → busca un paquete por la descripción.
11. `aptitude show paquete | less` → muestra información sobre un paquete.
12. `aptitude install paquete1 paquete2` ... → instala varios paquetes con sus dependencias y recomendaciones.
13. `aptitude -R install paquete` → instala un paquete con sus dependencias, pero sin las recomendaciones.
14. `aptitude why paquete` → lista las razones por las que se debería instalar el paquete.
15. `aptitude why-not paquete` → lista las razones por las que no se puede instalar el paquete.
16. `aptitude -rsvW install paquete` → simula la instalación de un paquete con sus dependencias y recomendaciones, detallando cada una.
17. `aptitude remove paquete` → desinstala un paquete.
18. `aptitude purge paquete` → desinstala un paquete y lo limpia de la cache.
19. `aptitude clean` → limpia la cache de paquetes.
20. `apt-file -xi search 'sql'` → busca todos los paquetes que contengan un archivo o directorio con la expresión *sql* (sin considerar mayúsculas o minúsculas).[3)]

Ver el contenido de un fichero

1. `cat file1` → ver los contenidos de un fichero comenzando desde la primera línea.

2. `tac file1` → ver los contenidos de un fichero comenzando desde la última línea.
3. `more file1` → ver el contenido de un fichero de manera paginada.
4. `less file1` → parecido al commando 'more' pero permite avanzar, retroceder, y buscar (compatible con algunos comandos de `vi`).
5. `head -2 file1` → ver las dos primeras líneas de un fichero (ó 10, si no se especifica la cantidad de líneas).
6. `tail -2 file1` → ver las dos últimas líneas de un fichero (ó 10, si no se especifica la cantidad de líneas).

Manipulación de texto

1. `cat file1 file2 … | command <> file1_in.txt_or_file1_out.txt` → sintaxis general para la manipulación de texto utilizando PIPE, STDIN y STDOUT.
2. `cat file1 | command( sed, grep, awk, grep, etc…) > result.txt` → sintaxis general para manipular un texto de un fichero y escribir el resultado en un fichero nuevo.
3. `cat file1 | command( sed, grep, awk, grep, etc…) >> result.txt` → sintaxis general para manipular un texto de un fichero y añadir resultado en un fichero existente.
4. `grep Aug /var/log/messages` → buscar palabras "Aug" en el fichero '/var/log/messages'.
5. `grep ^Aug /var/log/messages` → buscar palabras que comienzan con "Aug" en fichero '/var/log/messages'
6. `grep [0-9] /var/log/messages` → seleccionar todas las líneas del fichero '/var/log/messages' que contienen números.
7. `grep Aug -R /var/log/` → buscar la cadena "Aug" en el directorio '/var/log' y debajo.
8. `sed 's/string1/string2/g' ejemplo.txt` → reemplazar en ejemplo.txt todas las ocurrencias de "string1" con "string2"
9. `sed '/^$/d' ejemplo.txt` → eliminar todas las líneas en blanco desde el ejemplo.txt
10. `sed '/ *#/d; /^$/d' ejemplo.txt` → eliminar comentarios y líneas en blanco de ejemplo.txt
11. `echo 'ejemplo' | tr '[:lower:]' '[:upper:]'` → convertir "ejemplo" de minúsculas a mayúsculas.
12. `sed -e '1d' ejemplo.txt` → elimina la primera línea del fichero ejemplo.txt
13. `sed -n '/string1/p'` → visualizar solamente las líneas que contienen la palabra "string1".
14. `sed -r 's/(cadena1)(cadena2)/\2\1/g'` → utilizar expresiones regulares extendidas para intercambiar el orden de dos cadenas de texto, en todas las instancias que aparezcan.

Establecer caracter y conversión de ficheros

1. `dos2unix filedos.txt fileunix.txt` → convertir un formato de fichero texto desde MSDOS a UNIX.
2. `unix2dos fileunix.txt filedos.txt` → convertir un formato de fichero de texto desde UNIX a MSDOS.
3. `recode .. < page.txt > page.html` → convertir un fichero de texto en html.
4. `recode -l | more` → mostrar todas las conversiones de formato disponibles.

Análisis del sistema de ficheros

1. `badblocks -v /dev/hda1` → Chequear los bloques defectuosos en el disco hda1.
2. `fsck /dev/hda1` → reparar / chequear la integridad del fichero del sistema Linux en el disco hda1.
3. `fsck.ext2 /dev/hda1` → reparar / chequear la integridad del fichero del sistema ext 2 en el disco hda1.
4. `e2fsck /dev/hda1` → reparar / chequear la integridad del fichero del sistema ext 2 en el disco hda1.
5. `e2fsck -j /dev/hda1` → reparar / chequear la integridad del fichero del sistema ext 3 en el disco hda1.
6. `fsck.ext3 /dev/hda1` → reparar / chequear la integridad del fichero del sistema ext 3 en el disco hda1.
7. `fsck.vfat /dev/hda1` → reparar / chequear la integridad del fichero sistema fat en el disco hda1.
8. `fsck.msdos /dev/hda1` → reparar / chequear la integridad de un fichero del sistema dos en el disco hda1.
9. `dosfsck /dev/hda1` → reparar / chequear la integridad de un fichero del sistema dos en el disco hda1.

Formatear un sistema de ficheros

1. `mkfs /dev/hda1` → crear un fichero de sistema tipo Linux en la partición hda1.
2. `mke2fs /dev/hda1` → crear un fichero de sistema tipo Linux ext 2 en hda1.
3. `mke2fs -j /dev/hda1` → crear un fichero de sistema tipo Linux ext3 (periódico) en la partición hda1.
4. `mkfs -t vfat 32 -F /dev/hda1` → crear un fichero de sistema FAT32 en hda1.
5. `fdformat -n /dev/fd0` → formatear un disco flooply.
6. `mkswap /dev/hda3` → crear un fichero de sistema swap.

Partición de sistema swap

1. `mkswap /dev/hda3` → crear fichero de sistema swap.
2. `swapon /dev/hda3` → activando una nueva partición swap.
3. `swapon /dev/hda2 /dev/hdb3` → activar dos particiones swap.

Salvas

1. `dump -0aj -f /tmp/home0.bak /home` → hacer una salva completa del directorio '/home'.
2. `dump -1aj -f /tmp/home0.bak /home` → hacer una salva incremental del directorio '/home'.
3. `restore -if /tmp/home0.bak` → restaurando una salva interactivamente.
4. `rsync -rogpav --delete /home /tmp` → sincronización entre directorios.
5. `rsync -rogpav -e ssh --delete /home ip_address:/tmp` → rsync a través del túnel SSH.
6. `rsync -az -e ssh --delete ip_addr:/home/public /home/local` → sincronizar un directorio local con un directorio remoto a través de ssh y de compresión.
7. `rsync -az -e ssh --delete /home/local ip_addr:/home/public` → sincronizar un directorio remoto con un directorio local a través de ssh y de compresión.
8. `dd bs=1M if=/dev/hda | gzip | ssh user@ip_addr 'dd of=hda.gz'` → hacer una salva de un disco duro en un host remoto a través de ssh.
9. `dd if=/dev/sda of=/tmp/file1` → salvar el contenido de un disco duro a un fichero. (En este caso el disco duro es “sda” y el fichero “file1”).
10. `tar -Puf backup.tar /home/user` → hacer una salva incremental del directorio '/home/user'.
11. `tar -czv --exclude=/root/dir1/* -f /var/salvas/cfg_$(date +%F_%H%M).tgz /etc /root` → salvar los directorios /etc y /root (excluyendo el contenido del subdirectorio /root/dir1/) en un archivo comprimido, cuyo nombre contenga la fecha y hora actual.
12. `( cd /tmp/local/ && tar c . ) | ssh -C user@ip_addr 'cd /home/share/ && tar x -p'` → copiar el contenido de un directorio en un directorio remoto a través de ssh.
13. `( tar c /home ) | ssh -C user@ip_addr 'cd /home/backup-home && tar x -p'` → copiar un directorio local en un directorio remoto a través de ssh.
14. `tar cf - . | (cd /tmp/backup ; tar xf - )` → copia local conservando las licencias y enlaces desde un directorio a otro.
15. `find /home/user1 -name '*.txt' | xargs cp -av --target-directory=/home/backup/ --parents` → encontrar y copiar todos los ficheros con extensión '.txt' de un directorio a otro.
16. `find ~/ -type f -not -iname '*.mp*' -not \( -iregex '.*\.mozilla/.*' -o -iregex '.*\.thumbnails/.*' \) -prune -mtime 14 -print0 | xargs -0 tar -czf /var/salvas/myhome.tgz` → realizar una salva de todos los archivos de nuestro perfil de usuario modificados dentro de los últimos 14 días, exceptuando los archivos mp3, mpg y similares, y los directorios `.mozilla/` y `.thumbnails/`.
17. `find /var/log -name '*.log' | tar cv --files-from=- | bzip2 > log.tar.bz2` → encontrar todos los ficheros con extensión '.log' y hacer un archivo bzip.
18. `dd if=/dev/hda of=/dev/fd0 bs=512 count=1` → hacer una copia del MRB (Master Boot Record) a un disco floppy.

19. `dd if=/dev/fd0 of=/dev/hda bs=512 count=1` → restaurar la copia del MBR (Master Boot Record) salvada en un floppy.

CDROM

1. `cdrecord -v gracetime=2 dev=/dev/cdrom -eject blank=fast -force` → limpiar o borrar un cd regrabable.
2. `mkisofs /dev/cdrom > cd.iso` → crear una imagen iso de cdrom en disco.
3. `mkisofs /dev/cdrom | gzip > cd_iso.gz` → crear una imagen comprimida iso de cdrom en disco.
4. `mkisofs -J -allow-leading-dots -R -V “Label CD” -iso-level 4 -o ./cd.iso data_cd` → crear una imagen iso de un directorio.
5. `cdrecord -v dev=/dev/cdrom cd.iso` → quemar una imagen iso.
6. `gzip -dc cd_iso.gz | cdrecord dev=/dev/cdrom -` → quemar una imagen iso comprimida.
7. `mount -t udf,iso9660 -o loop cd.iso /mnt/iso` → montar una imagen iso.
8. `cd-paranoia -B` → llevar canciones de un cd a ficheros wav.
9. `cd-paranoia -- ”-3”` → llevar las 3 primeras canciones de un cd a ficheros wav.
10. `cdrecord --scanbus` → escanear bus para identificar el canal scsi.
11. `dd if=/dev/hdc | md5sum` → hacer funcionar un md5sum en un dispositivo, como un CD.
12. `eject -v` → expulsar un medio o disco extraíble, ofreciendo información adicional.

Trabajo con la red (LAN Y WIFI)

1. `ifconfig eth0` → mostrar la configuración de una interfaz de red Ethernet.
2. `ifup eth0` → activar la interfaz `eth0`.
3. `ifdown eth0` → deshabilitar la interfaz `eth0`.
4. `ifconfig eth0 192.168.1.1 netmask 255.255.255.0` → configurar una dirección IP.
5. `ifconfig eth0 promisc` → configurar `eth0` en modo promiscuo para obtener los paquetes (*sniffing*).
6. `dhclient eth0` → activar la interface 'eth0' en modo dhcp.
7. `route -n` → mostrar tabla de rutas.
8. `route add -net 0/0 gw IP_Gateway` → configurar entrada predeterminada.
9. `ip route show | grep default | awk {'print $3'}` → conocer la puerta de enlace predeterminada.
10. `route add -net 192.168.0.0 netmask 255.255.0.0 gw 192.168.1.1` → configurar ruta estática para buscar la red `192.168.0.0/16`.
11. `route del 0/0 gw IP_gateway` → eliminar la ruta estática.
12. `echo 1 > /proc/sys/net/ipv4/ip_forward` → activar el redireccionamiento de paquetes ip.
13. `hostname` → mostrar el nombre del host del sistema.

14. `host www.example.com` → buscar el nombre del host para resolver el nombre a una dirección ip (1).
15. `nslookup www.example.com` → buscar el nombre del host para resolver el nombre a una direcciòm ip y viceversa (2).
16. `ip link show` → mostar el estado de enlace de todas las interfaces.
17. `mii-tool eth0` → mostar el estado de enlace de `eth0`.
18. `ethtool eth0` → mostrar las estadísticas de la interfaz de red `eth0`.
19. `netstat -tup` → mostrar todas las conexiones de red activas y sus PID.
20. `netstat -tupl` → mostrar todos los servicios de escucha de red en el sistema y sus PID.
21. `netstat -punta` → mostrar todas las conexiones activas por dirección IP y puerto.
22. `tcpdump tcp port 80` → mostrar todo el tráfico HTTP.
23. `iwlist scan` → mostrar las redes inalámbricas.
24. `iwconfig eth1` → mostrar la configuración de una interfaz de red inalámbrica.
25. `whois www.example.com` → buscar en base de datos Whois.
26. `iftop -nNP -i eth0` → mostrar en tiempo real las conexiones abiertas en eth0 y su tasa de transferencia.
27. `sockstat` → mostrar información sobre las conexiones abiertas.
28. `arp-scan -l` → descubrir en la red las direcciones IP y MAC.
29. `nm-tool` → muestra configuración de red (en caso de usar Network Manager obtiene los).

Redes de Microsoft Windows (Samba)

1. `nbtscan ip_addr` → resolución de nombre de red bios.
2. `nmblookup -A ip_addr` → resolución de nombre de red bios.
3. `smbclient -L ip_addr/hostname` → mostrar acciones remotas de un host en windows.

Cortafuegos (iptables)

1. `iptables -t filter -L` → mostrar todas las cadenas de la tabla de filtro.
2. `iptables -t nat -L` → mostrar todas las cadenas de la tabla nat.
3. `iptables -t filter -F` → limpiar todas las reglas de la tabla de filtro.
4. `iptables -t nat -F` → limpiar todas las reglas de la tabla nat.
5. `iptables -t filter -X` → borrar cualquier cadena creada por el usuario.
6. `iptables -t filter -A INPUT -p tcp --dport telnet -j ACCEPT` → permitir las conexiones telnet para entar.
7. `iptables -t filter -A OUTPUT -p tcp --dport http -j DROP` → bloquear las conexiones HTTP para salir.
8. `iptables -t filter -A FORWARD -p tcp --dport pop3 -j ACCEPT` → permitir las conexiones POP a una cadena delantera.
9. `iptables -t filter -A INPUT -p tcp -m multiport --dports 80,443,8080 -m state --state NEW -m limit --limit 4/sec --limit-burst 8 -j ACCEPT` → establecer un límite de 4 peticiones por segundo de

nuevas conexiones, con posibles ráfagas ocasionales (útil para políticas de denegación por defecto).

10. `iptables -t filter -A INPUT -p tcp -m multiport --dports 80,443,8080 -m state --state ESTABLISHED,RELATED -m connlimit ! --conlimit-above 6 -j ACCEPT` → establecer un límite de 6 conexiones simultáneas por equipo a nuestro servidor web (útil para políticas de denegación por defecto).
11. `iptables -t filter -A INPUT -j LOG --log-prefix “DROP INPUT”` → registrando una cadena de entrada.
12. `iptables -t nat -A POSTROUTING -o eth0 -j MASQUERADE` → configurar un PAT (Puerto de traducción de dirección) en eth0, ocultando los paquetes de salida forzada. (Indicado para enlaces tipo ppp)
13. `iptables -t nat -A POSTROUTING -s 192.168.0.127 -o eth0 -j SNAT --to-source 169.158.158.169` → enrutar los paquetes desde 192.168.0.127 hacia otras redes por eth0 y asignarles un dirección ip externa. (Indicado para enlaces tipo ADSL)
14. `iptables -t nat -A PREROUTING -d 192.168.0.1 -p tcp -m tcp --dport 22 -j DNAT --to-destination 10.0.0.2:22` → redireccionar los paquetes diriguidos de un host a otro.
15. `iptables -t nat -S` → Listar todas las reglas activas en la tabla nat.
16. `iptables-save -c > archivo` → Salvar las reglas en un archivo (incluyendo los contadores de paquetes y bytes).
17. `iptables-restore -c < archivo` → Restaurar las reglas desde un archivo (incluyendo los contadores de paquetes y bytes).

Monitoreando y depurando

1. `top` → mostrar en tiempo real las tareas de linux.
2. `htop` → mostrar y gestionar las tareas con una interfaz amistosa.
3. `ps -eafw` → muestra las tareas Linux.
4. `ps -e -o pid,args --forest` → muestra las tareas Linux en un modo jerárquico.
5. `ps -o pid,cmd -ww -C wget` → listar todas las instancias del comando `wget` con sus argumentos.
6. `pstree` → mostrar un árbol sistema de procesos.
7. `pidof pppd` → mostrar el `pid` del proceso `pppd`.
8. `kill -9 ID_Processo` → forzar el cierre de un proceso y terminarlo.
9. `kill -1 ID_Processo` → forzar un proceso para recargar la configuración.
10. `killall Nombre_Proceso` → terminar un proceso por el nombre del comando y no por el ID.
11. `kill -HUP $(ps -A -o state,pid --no-header | grep -e '^[Zz]' | awk '{print $2}' | xargs)` → terminar todos los procesos *zombies*.
12. `lsof -p $$` → mostrar una lista de ficheros abiertos por procesos.
13. `lsof /home/user1` → muestra una lista de ficheros abiertos en un camino dado del sistema.

14. `strace -c ls >/dev/null` → mostrar las llamadas del sistema hechas y recibidas por un proceso.
15. `strace -f -e open ls >/dev/null` → mostrar las llamadas a la biblioteca.
16. `watch -n1 'cat /proc/interrupts'` → mostrar interrupciones en tiempo real.
17. `last reboot` → mostrar historial de reinicio.
18. `lsmod` → mostrar los módulos del kernel cargados.
19. `free -m` → muestra el estado de la RAM en megabytes.
20. `smartctl -A /dev/hda` → monitorear la fiabilidad de un disco duro a través de SMART.
21. `smartctl -i /dev/hda` → chequear si SMART está activado en un disco duro.
22. `tail -f /var/log/dmesg` → mostrar eventos inherentes al proceso de carga del kernel.
23. `tail -f /var/log/messages` → mostrar los eventos del sistema.
24. `multitail --follow-all /var/log/{dmesg,messages}` → mostrar dos registros de eventos en una misma pantalla.

Seguridad y Cifrado

1. `base64 /home/archivo > /home/archivo-codificado` → codifica “archivo” en 'base64' y lo guarda en /home
2. `base64 -d /home/archivo-codificado > /home/archivo` → decodifica “archivo-codificado” y lo guarda en /home
3. `openssl req -x509 -nodes -days 3650 -newkey rsa:1024 -out /etc/millave.crt -keyout /etc/millave.key` → crea un certificado auto-firmado para cifrar el tráfico web con SSL.
4. `htpasswd -c -m /etc/apache2/.htpasswd nombreusuario` → genera un archivo '.htpasswd' para proteger un sitio web con auntenticación.

Otros comandos útiles

1. `apropos palabraclave` → mostrar una lista de comandos que pertenecen a las palabras claves de un programa; son útiles cuando tú sabes qué hace tu programa, pero desconoces el nombre del comando.
2. `man ping` → mostrar las páginas del manual on-line; por ejemplo, en un comando ping, usar la opción '-k' para encontrar cualquier comando relacionado.
3. `man -t ping | ps2pdf - ping.pdf` → convertir las páginas del manual del comando ping en un archivo pdf (para lo cual es necesario haber instalado Ghostscript).
4. ``mkbootdisk --device /dev/fd0 `uname -r` `` → crear un floppy boteable.
5. `gpg -c file1` → codificar un fichero con guardia de seguridad GNU.
6. `gpg file1.gpg` → decodificar un fichero con Guardia de seguridad GNU.
7. `wget -r www.example.com` → descargar un sitio web completo.
8. `wget -c www.example.com/file.iso` → descargar un fichero con la posibilidad de parar la descargar y reanudar más tarde.

9. `echo 'wget -c www.example.com/files.iso' | at 09:00` → Comenzar una descarga a cualquier hora. En este caso empezaría a las 9 horas.
10. `ldd /usr/bin/ssh` → mostrar las bibliotecas compartidas requeridas por el programa ssh.
11. `alias hh='history`' → colocar un alias para un commando. En este caso, para invocar el historial con `hh`.
12. `chsh` → cambiar el comando Shell.
13. `chsh --list-shells` → es un comando adecuado para saber si tienes que hacer remoto en otra terminal.
14. `who -a` → mostrar quien está registrado, e imprimir hora del último sistema de importación, procesos muertos, procesos de registro de sistema, procesos activos producidos por init, funcionamiento actual y últimos cambios del reloj del sistema.
15. `echo "128*1024*1024" | bc` → calcular desde la consola el tamaño en bytes de 128 MiB.
16. `sudo !!` → ejecutar como superusuario el último comando tecleado.
17. `clear` → limpiar la pantalla.
18. `uncomando > archivodesalida.txt 2>&1` → ejecuta un comando y redirige la salida a un archivo, combinando en este tanto STDOUT como STDERR.
19. `uncomando > archivodesalida.txt 2> archivoerrores.txt` → ejecuta un comando, redirige la salida (STDOUT) a un archivo, y los errores (STDERR) a otro.
20. `uncomando | tee archivodesalida.txt` → ejecuta un comando, muestra la salida en la pantalla y simultáneamente la escribe a un archivo.

[1)] Es preciso ejecutarlo con privilegios de root.

[2)] Para conocer el límite de argumentos que `xargs` admite, puede ejecutarse el comando `echo | xargs --show-limits`

[3)] Para que este comando funcione, después de instalado el paquete `apt-file` es necesario invocar el comando `apt-file update`.